

Estado Actual y Perspectivas de la Investigación en Líneas de Productos de Software en la Pontificia Universidad Javeriana de Cali

Nombre1 Apellido1*, Nombre2 Apellido2*, Nombre3 Apellido3*

*Departamento de Electrónica y Ciencias de la Computación,
Pontificia Universidad Javeriana, Cali, Colombia

Resumen—Las líneas de productos de software son una alternativa para gestionar de manera eficiente el desarrollo de familias de productos, reutilizando los elementos comunes y administrando la variabilidad. Sin embargo el desarrollo de una línea de productos puede ser complejo, por ejemplo en la definición de los modelos, la selección de configuraciones de los productos y el proceso de ensamblaje. Por ello se requieren metodologías y herramientas que hagan posible el desarrollo efectivo de líneas de productos de software. Se presentan las líneas de investigación actuales en el Grupo Destino relacionadas con la gestión de la variabilidad y la reutilización de software, tanto en el aspecto de modelado como en los de diseño y construcción de productos a través de líneas de productos de software. Finalmente, se hace explícita la intención del grupo de trabajar con empresas del sector de software que estén interesadas en poner prueba los desarrollos actuales en el tema en un dominio específico de su interés.

Keywords—*Líneas de productos de software, defectos, modelos de variabilidad, configuración, ensamblaje*

I. INTRODUCCIÓN

En la industria de software, la mayoría de las empresas desarrollan productos para satisfacer necesidades en unos pocos dominios de aplicación. Por consiguiente, generalmente los productos de una empresa tienen muchas características y elementos comunes. Entonces, las prácticas de reutilización de componentes pueden acortar los tiempos de desarrollo de estos productos y mejorar la competitividad de las empresas. Sin embargo, para que la reutilización sea efectiva se requieren metodologías que, apoyadas en la tecnología, ayuden a administrar los procesos de reutilización durante el ciclo de desarrollo del producto.

Las Líneas de Productos de Software (LPS) son un paradigma de producción que permite desarrollar y administrar, de manera eficiente, productos que tienen elementos comunes y variables. Así, a través de la reutilización de elementos las LPS ofrecen beneficios como menores tiempos de desarrollo, menores costos, mayor calidad y la flexibilidad para ofrecer a los usuarios productos que se ajusten a sus necesidades.

Son varias las etapas que incluyen las metodologías de desarrollo de líneas de productos de software. Entre ellas: La definición de modelos de variabilidad que especifican, para una familia de productos, las características comunes y variables, sus relaciones y las restricciones sobre las mismas. El diseño y desarrollo de los artefactos que soportan la línea de productos. La selección de elementos que satisfacen un

conjunto particular de requerimientos, llamada configuración. Y, finalmente, el ensamblaje de los elementos para obtener un producto en particular.

De otra parte, el desarrollo de líneas de productos puede ser complejo en varios sentidos: Primero, los modelos de variabilidad pueden incluir un gran número de características y restricciones, por ello son susceptibles de incluir errores semánticos. Además, la cantidad de opciones de configuración puede ser tan grande que no sea posible hacer una lista que incluya todas las posibilidades, por lo tanto se hace necesario guiar las decisiones que se toman durante el proceso de configuración. Finalmente, el ensamblaje puede incluir elementos implementados con múltiples herramientas y bajo diversos paradigmas, lo cual hace necesario que el proceso de desarrollo siga una estrategia de ensamblaje de productos. Entonces, desarrollar metodologías y herramientas que den solución a estas y otras dificultades son temas importantes de investigación en el área de líneas de producto de software.

El Grupo de Investigación Destino de la Pontificia Universidad Javeriana está incursionando en el área de Líneas de Producto de Software a través de dos líneas de trabajo que se enfocan en aportar soluciones a las dificultades antes mencionadas. Nuestro objetivo es contribuir al desarrollo del paradigma LPS, de manera que pueda ser efectiva y ampliamente adoptado en la industria del software, en particular a nivel nacional y regional. Este artículo presenta los retos y perspectivas planteados para los proyectos que se están desarrollando en dichas líneas.

II. LÍNEAS DE INVESTIGACIÓN EN LPS EN EL GRUPO DESTINO

La producción intensiva de software requiere del uso de herramientas conceptuales y de artefactos que permitan garantizar la calidad, al tiempo que se acelera el proceso de construcción de aplicaciones. La gestión de la variabilidad está en el corazón de este propósito y es por ese motivo que en la Pontificia Universidad Javeriana se viene consolidando una línea de investigación en esta temática. Las líneas de productos de software son un paradigma de producción de software que promete materializar las consecuencias más importantes y positivas de una buena gestión de la variabilidad. Ellas incorporan aspectos de modelado que permiten capturar de manera rigurosa las características del dominio de aplicación, y también aspectos relacionados con la configuración y el ensamblaje de productos específicos, a partir de artefactos

previamente contruidos y reutilizables que materialicen en el producto las características requeridas por el cliente. Nuestros intereses de investigación actuales, en el tema de líneas de productos de software, son dos: aportar en el área de modelado, investigando sobre métodos para garantizar la calidad de los modelos de variabilidad a través de la detección de errores semánticos y su corrección; y la adopción del paradigma de líneas de productos de software con casos de aplicación reales. A continuación se detallan ambas líneas de trabajo.

II-A. Calidad de los modelos de variabilidad. ¿Cómo identificar de una manera genérica los defectos, sus causas y correcciones en los modelos de variabilidad?

El dominio de aplicación de una línea de productos de software se representa por medio de un modelo de variabilidad, que además sirve como un punto de comunicación entre los usuarios finales y los desarrolladores de la LPS [1]. Este modelo permite identificar por ejemplo, cuáles características son comunes a todos los productos de la línea de productos, cuáles características son variables; cuáles características deben ir juntas en todos los productos y cuáles características no pueden ir al mismo tiempo en un producto [1].

El modelo de variabilidad se diseña en una de las primeras etapas del desarrollo de la línea de productos y es un insumo importante para el resto del proceso de la ingeniería de línea de productos. Sin embargo, a medida que aumenta la complejidad del modelo de variabilidad, es posible que quienes lo elaboran introduzcan en él, sin intención, defectos semánticos. Los defectos semánticos son imperfecciones que afectan la capacidad del modelo de variabilidad para expresar de manera correcta el dominio que representa la línea de productos [2].

Rincón *et al.* [3], [4] proponen, para modelos de variabilidad representados con la notación de los modelos de características (en inglés *Feature Models* [1]), un método que identifica defectos semánticos y sugiere correcciones. Para ello, toman de la programación por restricciones el concepto de MCSes (*Minimal Correction Subsets*) y lo aplican para identificar las correcciones de los defectos semánticos encontrados. Específicamente, los MCSes corresponden a las dependencias que deben ser eliminadas de los modelos de variabilidad para solucionar al menos un defecto.

Antes de entregar los resultados al usuario final, el método traduce a lenguaje natural las correcciones identificadas para cada defecto. De esta manera se obtienen los defectos semánticos de los modelos de variabilidad y las correcciones para cada defecto en un lenguaje fácilmente entendible por quienes diseñan los modelos.

Los MCSes se han usado para identificar correcciones de defectos semánticos en modelos de características. Sin embargo, los modelos de variabilidad pueden ser expresados en otras notaciones como OVM [5], o Dopler [6], que no son consideradas en el trabajo de Rincón *et al.* [3], [4]. Así mismo, es necesario evaluar el método propuesto considerando aspectos como la satisfacción del usuario final, la utilidad de la solución, la escalabilidad, entre otros. La línea de trabajo orientada hacia la calidad de los modelos de variabilidad busca considerar estos aspectos con el fin de generalizar el método propuesto por Rincón *et al.* [3], [4] para la identificación de defectos, causas y sus correcciones.

II-B. Adopción del paradigma de líneas de productos de software. ¿Cómo apoyar el proceso de configuración y ensamblaje de productos de software para facilitar la transferencia de esta tecnología a la industria?

Los avances que se van haciendo en el aspecto teórico de las líneas de productos de software, deben validarse en casos concretos de utilización que hagan evidentes las fortalezas y debilidades de las teorías propuestas. Con este propósito en mente, se está iniciando una línea de trabajo que apunta al desarrollo de herramientas computacionales de apoyo a los procesos de configuración y ensamblaje, que permitan la construcción de líneas de producto concretas para algunos dominios utilizados como caso de estudio. Esta línea de investigación tiene como tareas concretas: la definición de buenas prácticas para la representación y depuración de los modelos de variabilidad, y el desarrollo de herramientas computacionales de apoyo a la realización de los procesos de configuración y ensamblaje de productos.

La tarea relacionada con la definición de buenas prácticas para el modelado de dominios inicia con la representación de dominios particulares e incluye las siguientes actividades: la evaluación de diferentes dominios para seleccionar algunos sobre los cuales desarrollar LPSs; la definición de los requerimientos; la definición del modelo conceptual de la LPS y el modelo de variabilidad; la aplicación de métodos de verificación a los modelos de variabilidad; y finalmente, la representación del modelo de variabilidad en un lenguaje formal, que sirva como insumo para el proceso de configuración. A la par que se llevan a cabo estas etapas de modelado, se ponen a prueba las notaciones, las herramientas y las representaciones, con el fin de deducir buenas prácticas de modelado.

En la tarea relacionada con el desarrollo de herramientas de apoyo, se ha definido un proyecto para elaborar un conjunto de prácticas, artefactos y criterios genéricos (framework) que apoye la configuración guiada de productos y el ensamblaje de componentes en líneas de productos de software. Estos aspectos, la configuración y el ensamblaje de productos de software, son fundamentales en nuestros intereses de investigación por lo que se explican más detalladamente a continuación.

1) Configuración de productos: En la ingeniería de líneas de productos el desarrollo de una línea de productos se divide en dos procesos: la **ingeniería de dominio** y la **ingeniería de aplicación**. En la ingeniería de dominio se analiza el área de aplicación en la que se desea crear la línea de productos de software, se identifican los elementos comunes y variables de dicha línea, y se diseñan y construyen los artefactos (*core assets*) que la soportan [5]. Por su parte, en la ingeniería de aplicación, los usuarios seleccionan características de los productos a partir de las cuales éstos se configuran y ensamblan [5]. El propósito de las líneas de productos de software es desarrollar artefactos durante la ingeniería del dominio que luego permitan obtener productos individuales durante la ingeniería de aplicación [7].

Una **configuración** es una selección de elementos que deben satisfacer tanto los requerimientos del usuario como las restricciones del dominio de la línea de productos [8]. Por esta razón, configurar un producto de una línea de productos debe ser un proceso guiado paso a paso que satisfaga tanto las necesidades del usuario como las restricciones propias del do-

minio de la LPS [2]. Además, el proceso de configuración debe satisfacer algunas características no funcionales relacionadas con el tiempo de respuesta, el orden de selección, la adaptación a un entorno cambiante, y la selección de características óptimas de acuerdo a los intereses del usuario, entre otras [9]. El proceso de configuración es exitoso si logra construir una configuración válida. Una configuración válida no tiene conflictos entre sus elementos y representa los requerimientos de un producto que pertenece a la línea de productos.

Nuestra investigación busca definir técnicas de apoyo a la configuración basadas en los requerimientos de los usuarios, la información de productos previamente configurados y las restricciones relacionadas con el modelo de variabilidad. Adicionalmente, se pretende definir las prácticas y los criterios que se deben seguir para facilitar la configuración guiada de los productos de software y desarrollar herramientas que permitan a los usuarios configurar productos de una LPS.

2) *Ensamblaje de componentes*: Una vez se configuran los productos de la línea de productos de software, es necesario establecer mecanismos para construir el producto de software configurado [10]. Este proceso, conocido como **ensamblaje**, debe además minimizar el esfuerzo necesario para obtener cada producto en particular [11].

Otro propósito de nuestra investigación es definir criterios y prácticas para la elaboración de artefactos reutilizables que puedan ser ensamblados de manera automática o semiautomática para generar productos que hagan parte de una LPS. Adicionalmente, se busca desarrollar criterios para la definición de la arquitectura de los productos y para la selección de las tecnologías con las que se implementan los componentes reutilizables. Por último, se busca definir prácticas y criterios para el ensamblaje de los componentes de software e implementar estrategias para relacionar las elecciones hechas por el usuario en la configuración con componentes específicos de la LPS.

III. RELACIÓN CON EL SECTOR INDUSTRIAL

Nuestro grupo de investigación Destino está interesado en construir colaboraciones con empresas de desarrollo de software que quieran aportar algún dominio específico para la validación de los esquemas de modelado propuestos y la prueba de las herramientas desarrolladas. De momento, se está trabajando en el dominio de las aplicaciones biométricas con la empresa XXX. Además, se está evaluando el dominio de las aplicaciones de apoyo a la rehabilitación auditiva que es un tema en el cual el grupo ha venido haciendo desarrollos en colaboración con el Instituto para Niños Ciegos y Sordos del Valle del Cauca. Es claro que entre mayor diversidad exista entre los dominios explorados, se podrá obtener una información más completa sobre las fortalezas y debilidades de los modelos, prácticas y herramientas desarrolladas para la construcción de líneas de productos de software. De esta evaluación surgirán también necesidades concretas aun no identificadas que darán origen a nuevas iniciativas de investigación y desarrollo.

IV. CONCLUSIONES

Las actuales prioridades de investigación en el campo de la ingeniería de software en nuestro grupo de investigación van

en dos líneas: una es el desarrollo de herramientas teóricas que permitan facilitar el modelado riguroso de los dominios de aplicación, su depuración y validación. La otra es el desarrollo de herramientas computacionales de apoyo a los procesos de configuración y ensamblaje de productos en una línea de productos de software, así como la puesta a prueba de todo el sistema en dominios reales de aplicación, ojalá con la colaboración de empresas de software interesadas en la transferencia de este conocimiento.

AGRADECIMIENTOS

Esta línea de trabajo puede seguir desarrollándose gracias a la financiación recibida de la Pontificia Universidad Javeriana para el año 2015.

REFERENCIAS

- [1] K. C. Kang, S. G. Cohen, J. A. Hess, W. E. Novak, and S. P. Peterson, "Feasibility Study Feature-Oriented Domain Analysis (FODA). Technical Report," Software Engineering Institute, Carnegie Mellon University, Tech. Rep., 1990.
- [2] C. Salinesi and R. Mazo, "Defects in Product Line Models and how to identify them," in *Software Product Line - Advanced Topic*, A. Elfaki, Ed. InTech, 2012, ch. 5, pp. 97–122.
- [3] L. Rincón, G. Giraldo, R. Mazo, C. Salinesi, and D. Díaz, "Método para Identificar Correcciones de Defectos en Modelos de Características que representan Líneas de Productos," in *Proceedings of the 40th Latin American Computing Conference*, Montevideo, 2014.
- [4] L. F. Rincón Pérez, G. L. Giraldo Gómez, R. Mazo, C. Salinesi, and D. Díaz, "Subconjuntos Mínimos de Corrección para explicar características muertas en Modelos de Líneas de Productos. El caso de los Modelos de Características," in *Proceedings of the 8th Colombian Computer Conference (CCC)*, Armenia-Colombia, 2013.
- [5] K. Pohl, G. Böckle, and F. J. van Der Linden, *Software Product Line Engineering: Foundations, Principles and Techniques*. Springer-Verlag New York, Inc., 2005.
- [6] D. Dhungana, R. Rabiser, P. Grünbacher, K. Lehner, and C. Federspiel, "DOPLER: An Adaptable Tool Suite for Product Line Engineering," in *Proceedings of the 11th International Software Product Line Conference*, ser. SPLC'07, Kyoto-Japan, 2007, pp. 151–152.
- [7] R. Rabiser, P. OLeary, and I. Richardson, "Key activities for product derivation in software product lines," *Journal of Systems and Software*, vol. 84, no. 2, pp. 285–300, Feb. 2011. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0164121210002700>
- [8] J. White, D. Benavides, D. Schmidt, P. Trinidad, B. Dougherty, and A. Ruiz-Cortés, "Automated diagnosis of feature model configurations," *Journal of Systems and Software*, vol. 83, no. 7, pp. 1094–1107, Jul. 2010.
- [9] R. Mazo, C. Dumitrescu, C. Salinesi, and D. Díaz, "Recommendation Heuristics for Improving Product Line Configuration Processes," in *Recommendation Systems in Software Engineering*, 2014, pp. 511–537.
- [10] S. Deelstra, M. Sinnema, and J. Bosch, "Product derivation in software product families: a case study," *Journal of Systems and Software*, vol. 74, no. 2, pp. 173–194, Jan. 2005. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0164121203003121>
- [11] P. Clements and L. M. Northrop, *Software Product Lines: Practices and Patterns*, 1st ed. Addison-Wesley Professional, 2001.